NXP USA Inc. - T2081NSE7TTB Datasheet

Welcome to E-XFL.COM

Understanding Embedded - Microprocessors

Embedded microprocessors are specialized computing chips designed to perform specific tasks within an embedded system. Unlike general-purpose microprocessors found in personal computers, embedded microprocessors are tailored for dedicated functions within larger systems, offering optimized performance, efficiency, and reliability. These microprocessors are integral to the operation of countless electronic devices, providing the computational power necessary for controlling processes, handling data, and managing communications.

Applications of Embedded - Microprocessors

Embedded microprocessors are utilized across a broad spectrum of applications, making them indispensable in

Details	
Product Status	Obsolete
Core Processor	PowerPC e6500
Number of Cores/Bus Width	4 Core, 64-Bit
Speed	1.8GHz
Co-Processors/DSP	-
RAM Controllers	DDR3, DDR3L
Graphics Acceleration	No
Display & Interface Controllers	-
Ethernet	1Gbps (8), 2.5Gbps (4), 10Gbps (4)
SATA	SATA 3Gbps (2)
USB	USB 2.0 + PHY (2)
Voltage - I/O	-
Operating Temperature	0°C ~ 105°C (TA)
Security Features	Boot Security, Cryptography, Secure Fusebox, Secure Debug, Volatile key Storage
Package / Case	780-FBGA, FCBGA
Supplier Device Package	780-FCPBGA (23x23)
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/t2081nse7ttb

3 Application examples

This chip is well-suited for applications that are highly compute-intensive, I/O-intensive, or both.

3.1 1U security appliance

This figure shows a 1U security appliance built around a single SoC. The QorIQ DPAA accelerates basic packet classification, filtering, and packet queuing, while the crypto accelerator, regex accelerator, and compression/decompression accelerator perform high throughput content processing. The high single threaded and aggregate DMIPS of the core CPUs provide the processing horsepower for complex classification and flow state tracking required for proxying applications as well as heuristic traffic analysis and policy enforcement.

The SoC's massive integration significantly reduces system BOM cost. SATA hard drives connect directly to the SoC's integrated controllers, and an Ethernet switch is only required if more than eight 1 GE ports or 4 10 GE ports are required. The SoC supports PCIe and Serial RapidIO for expansion.

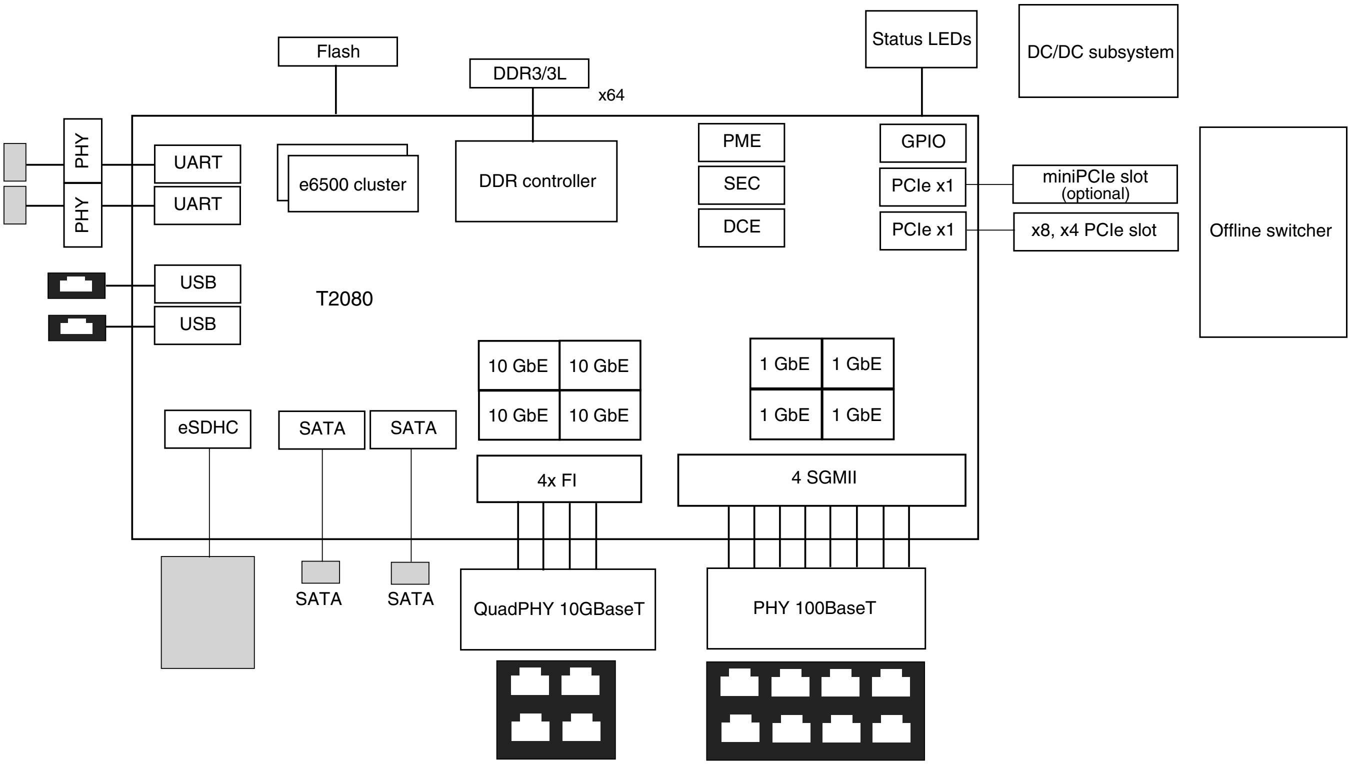

Flash
DDR3/3L
x64
Status LEDs
DC/DC subsystem
PHY
PHY
UART
UART
e6500 cluster
DDR controller
PME
SEC
DCE
GPIO
PCIe x1
PCIe x1
miniPCIe slot (optional)
x8, x4 PCIe slot
Offline switcher
USB
USB
T2080
10 GbE
10 GbE
10 GbE
10 GbE
1 GbE
1 GbE
1 GbE
1 GbE
eSDHC
SATA
SATA
4x FI
4 SGMII
SATA
SATA
QuadPHY 10GBaseT
PHY 100BaseT

Figure 1. SoC 1U security appliance

3.2 Radio node controller

Some of the more demanding packet-processing applications are found in the realm of wireless infrastructure. These systems have to interwork between wireless link layer protocols and IP networking protocols. Wireless protocol complexity is high, and includes scheduling, retransmission, and encryption with algorithms specific to cellular wireless access networks. Connecting to the IP network offers wireless infrastructure tremendous cost savings, but introduces all the security threats found in the IP world. The chip's network and peripheral interfaces provide it with the flexibility to connect to DSPs, and to wireless link layer framing ASICs/FPGAs . While the Data Path Acceleration Architecture offers encryption acceleration for both wireless and IP networking protocols, in addition to packet filtering capability on the IP networking side, multiple virtual CPUs may be dedicated to data path processing in each direction.

3.3 Intelligent network adapter

The exact form factor of this card may vary but the concepts are similar. A chip is placed on a small form factor card with an x8 PCIe connector and multiple 10 G Ethernet ports. This card is then used as inline accelerator that provides both line rate networking and intelligent programmable offload from a host processor subsystem in purpose built appliances and servers. This figure shows an example of a T2080 built as a PCI Express form-factor supporting virtualization through SR-IOV with quad 10 G physical networking interfaces.

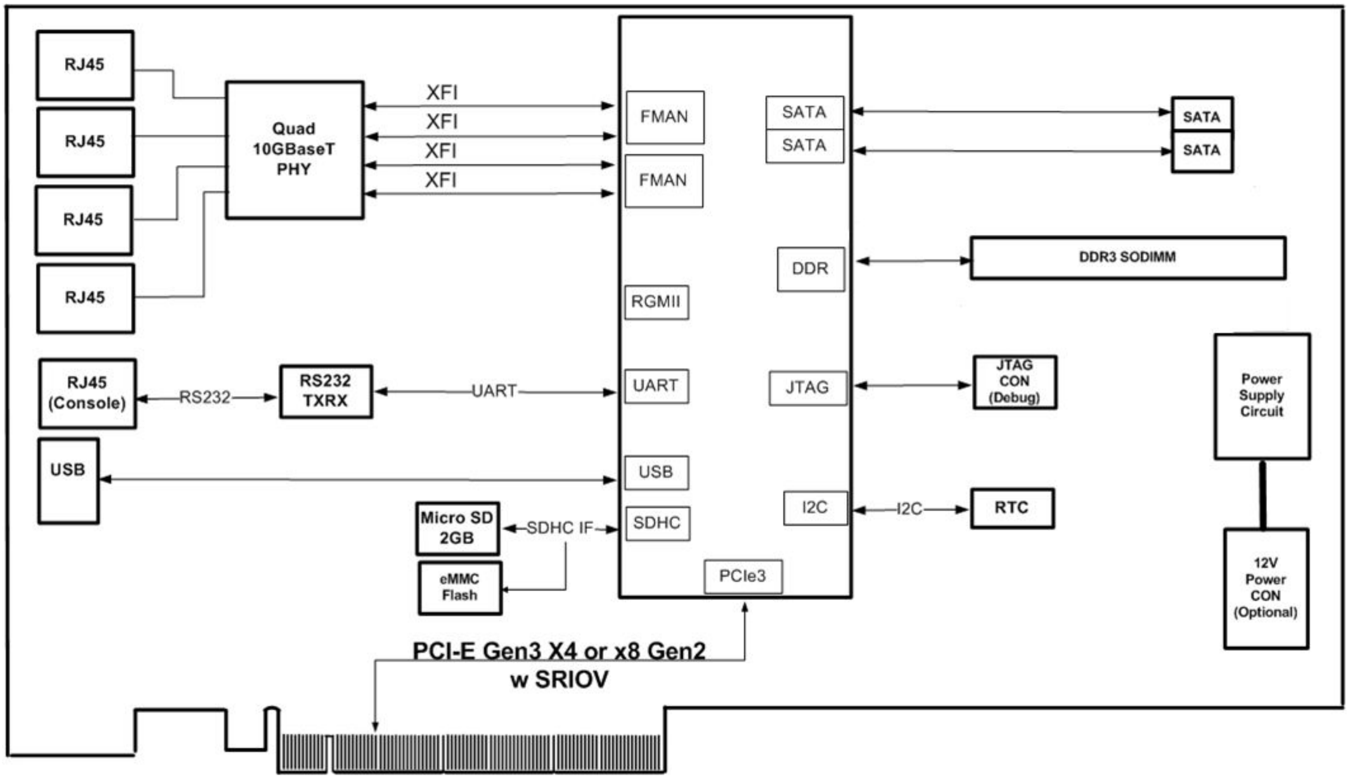

RJ45
RJ45
RJ45
RJ45
Quad 10GBaseT PHY
XFI
XFI
XFI
XFI
FMAN
FMAN
SATA
SATA
SATA
SATA
DDR
DDR3 SODIMM
RGMII
RJ45 (Console)
RS232
RS232 TXRX
UART
UART
JTAG
JTAG CON (Debug)
Power Supply Circuit
USB
USB
Micro SD 2GB
SDHC IF
SDHC
I2C
I2C
RTC
eMMC Flash
PCIe3
12V Power CON (Optional)
PCI-E Gen3 X4 or x8 Gen2 w SRIOV

Figure 2. Intelligent network adapter

4 Chip features

This section describes the key features and functionalities of the chip.

4.1 Block diagram

This figure shows the major functional units within the chip.

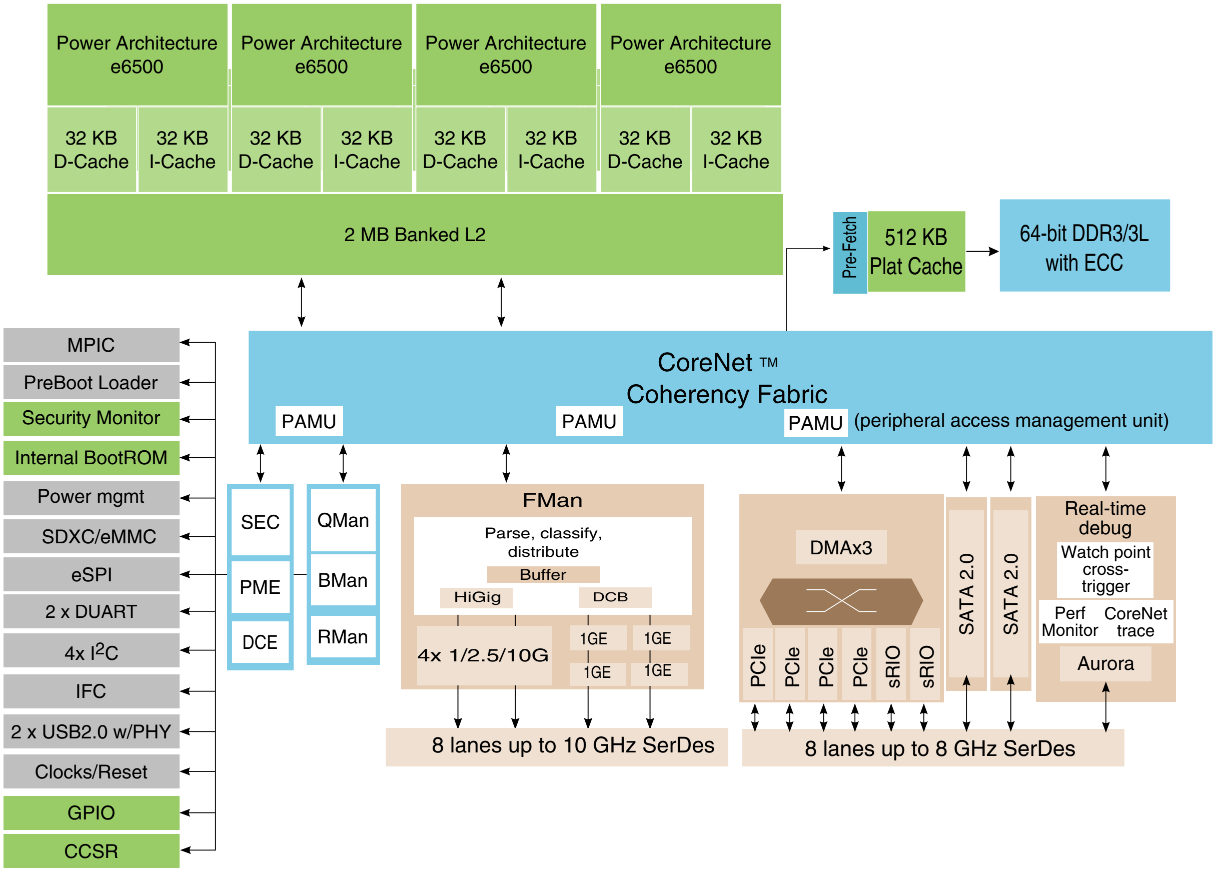

Power Architecture e6500
Power Architecture e6500
Power Architecture e6500
Power Architecture e6500
32 KB D-Cache
32 KB I-Cache
32 KB D-Cache
32 KB I-Cache
32 KB D-Cache
32 KB I-Cache
32 KB D-Cache
32 KB I-Cache
2 MB Banked L2
Pre-Fetch
512 KB Plat Cache
64-bit DDR3/3L with ECC
MPIC
PreBoot Loader
Security Monitor
Internal BootROM
Power mgmt
SDXC/eMMC
eSPI
2 x DUART
4x I²C
IFC
2 x USB2.0 w/PHY
Clocks/Reset
GPIO
CCSR
CoreNet ™ Coherency Fabric
PAMU
PAMU
PAMU (peripheral access management unit)
SEC
PME
DCE
QMan
BMan
RMan
FMan
Parse, classify, distribute
Buffer
HiGig
DCB
4x 1/2.5/10G
1GE
1GE
1GE
1GE
8 lanes up to 10 GHz SerDes
DMAx3
PCIe
PCIe
PCIe
PCIe
sRIO
sRIO
SATA 2.0
SATA 2.0
Real-time debug
Watch point cross-trigger
Perf Monitor
CoreNet trace
Aurora
8 lanes up to 8 GHz SerDes

Figure 3. T2080 block diagram

4.2 Features summary

This chip includes the following functions and features:

- 4, dual-threaded e6500 cores built on Power Architecture® technology sharing a 2 MB L2 cache
  - Up to 1.8 GHz with 64-bit ISA support (Power Architecture v2.06-compliant)
- 512 KB CoreNet platform cache (CPC)
- Hierarchical interconnect fabric
  - CoreNet fabric supporting coherent and non-coherent transactions with prioritization and bandwidth allocation amongst CoreNet end-points
  - Queue Manager (QMan) fabric supporting packet-level queue management and quality of service scheduling
- One 32-/64-bit DDR3/3L SDRAM memory controllers with ECC and interleaving support
  - Memory pre-fetch engine
- Data Path Acceleration Architecture (DPAA) incorporating acceleration for the following functions:
  - Packet parsing, classification, and distribution (Frame Manager)
  - Queue management for scheduling, packet sequencing, and congestion management (Queue Manager)
  - Hardware buffer management for buffer allocation and de-allocation (BMan)
  - Cryptography acceleration (SEC 5.2) at up to 10 Gbps
  - RegEx Pattern Matching Acceleration (PME 2.1) at up to 10 Gbps
  - Decompression/Compression Acceleration (DCE) at up to 17.5 Gbps
  - DPAA chip-to-chip interconnect via RapidIO Message Manager (RMAN)
- 16 SerDes lanes at up to 10.3125 GHz
- Eight Ethernet interfaces, supporting combinations of the following:

    - Up to four 10 Gbps Ethernet MACs
    - Up to eight 1 Gbps Ethernet MACs
    - Up to four 2.5 Gbps Ethernet MACs
- High-speed peripheral interfaces
    - Four PCI Express controllers (two support PCIe 2.0 and two support PCIe 3.0)
    - Two Serial RapidIO 2.0 controllers/ports running at up to 5 GHz with Type 11 messaging and Type 9 data streaming support
- Additional peripheral interfaces
    - Two serial ATA (SATA 2.0) controllers
    - Two high-speed USB 2.0 controllers with integrated PHY
    - Enhanced secure digital host controller (SD/SDXC/eMMC)
    - Enhanced serial peripheral interface (eSPI)
    - Four I2C controllers
    - Four 2-pin UARTs or two 4-pin UARTs
    - Integrated Flash controller supporting NAND and NOR flash
- Three eight-channel DMA engines
- Support for hardware virtualization and partitioning enforcement
- QorIQ Platform's Trust Architecture 2.0

4.3 Critical performance parameters

This table lists key performance indicators that define a set of values used to measure SoC operation.

Table 1. Critical performance parameters

Indicator	Values(s)
Top speed bin core frequency	1.8 GHz
Maximum memory data rate	2133 MHz (DDR3)[1], 1866.67 MHz for DDR3L • 1.5 V for DDR3 • 1.35 V for DDR3L
Integrated flash controller (IFC)	1.8 V
Operating junction temperature range Extended temperature version	0-105° C -40–105° C
Package	896-pin, flip-chip plastic ball grid array (FC-PBGA), 25 x 25mm

1. Conforms to JEDEC standard

4.4 Core and CPU clusters

This chip offers four, high-performance 64-bit Power Architecture Book E-compliant cores. Each CPU core supports two hardware threads, which software views as a virtual CPU.

This table shows the computing metrics the core supports.

Table 2. Power architecture metrics

Metric	Per core	Full device
DMIPS	10,800	43,200
Single-precision GFLOPs	14.4	72

Table continues on the next page...

    - Provides system software with an efficient means to move data and perform cache operations between two disjoint address spaces
    - Eliminates the need to copy data from a source context into a kernel context, change to destination address space, then copy the data to the destination address space or alternatively to map the user space into the kernel address space

The arrangement of cores into clusters with shared L2 caches is part of a major re-architecture of the QorIQ cache hierarchy. Details of the banked L2 are provided below.

- 2 MB cache with ECC protection (data, tag, & status)
- 64-byte cache line size
- 16 way, set associative
    - Ways in each bank can be configured in one of several modes
    - Flexible way partitioning per vCPU
        - I-only, D-only, or unified
- Supports direct stashing of datapath architecture data into L2

4.5 Inverted cache hierarchy

From the perspective of software running on an core vCPU, the SoC incorporates a 2-level cache hierarchy. These levels are as follows:

- Level 1: Individual core 32 KB Instruction and Data caches
- Level 2: Locally banked 2 MB cache (configurably shared by other vCPUs in the cluster)

Therefore, the CPC is not intended to act as backing store for the L2s. This allows the CPCs to be dedicated to the non-CPU masters in the SoC, storing DPAA data structures and IO data that the CPUs and accelerators will most likely need.

Although the SoC supports allocation policies that would result in CPU instructions and in data being held in the CPC (CPC acting as vCPU L3), this is not the default. Because the CPC serves fewer masters, it serves those masters better, by reducing the DDR bandwidth consumed by the DPAA and improving the average latency.

4.6 CoreNet fabric and address map

As Freescale's next generation front-side interconnect standard for multicore products, the CoreNet fabric provides the following:

- A highly concurrent, fully cache coherent, multi-ported fabric
- Point-to-point connectivity with flexible protocol architecture allows for pipelined interconnection between CPUs, platform caches, memory controllers, and I/O and accelerators at up to 800 MHz
- The CoreNet fabric has been designed to overcome bottlenecks associated with shared bus architectures, particularly address issue and data bandwidth limitations. The chip's multiple, parallel address paths allow for high address bandwidth, which is a key performance indicator for large coherent multicore processors.
- Eliminates address retries, triggered by CPUs being unable to snoop within the narrow snooping window of a shared bus. This results in the chip having lower average memory latency.

This chip's 40-bit, physical address map consists of local space and external address space. For the local address map, 32 local access windows (LAWs) define mapping within the local 40-bit (1 TB) address space. Inbound and outbound translation windows can map the chip into a larger system address space such as the RapidIO or PCIe 64-bit address environment. This functionality is included in the address translation and mapping units (ATMUs).

4.7 DDR memory controller

The chip offers a single DDR controller supporting ECC protected memories. This DDR controller operates at up to 2133 MHz for DDR3, and, in more power-sensitive applications, up to 1866.667 MHz for DDR3L. Some key DDR controller features are as follows:

- Support x8 and x16 memory widths
  - Programmable support for single-, dual-, and quad-ranked devices and modules
  - Support for both unbuffered and registered DIMMs
  - 4 chip-selects
  - 40-bit address support, up to 1 TB memory
- The SoC can be configured to retain the currently active SDRAM page for pipelined burst accesses. Page mode support of up to 64 simultaneously open pages can dramatically reduce access latencies for page hits. Depending on the memory system design and timing parameters, page mode can save up to ten memory clock cycles for subsequent burst accesses that hit in an active page.
- Using ECC, the SoC detects and corrects all single-bit errors and detects all double-bit errors and all errors within a nibble.
- Upon detection of a loss of power signal from external logic, the DDR controller can put compliant DDR SDRAM DIMMs into self-refresh mode, allowing systems to implement battery-backed main memory protection.
- In addition, the DDR controller offers an initialization bypass feature for use by system designers to prevent re-initialization of main memory during system power-on after an abnormal shutdown.
- Support active zeroization of system memory upon detection of a user-defined security violation.

4.7.1 DDR bandwidth optimizations

Multicore SoCs are able to increase CPU and network interface bandwidths faster than commodity DRAM technologies are improving. As a result, it becomes increasingly important to maximize utilization of main memory interfaces to avoid a memory bottleneck. The SoC's DDR controller Freescale-developed IP, optimized for the QorIQ SoC architecture, with the goal of improving DDR bandwidth utilization by fifty percent when compared to first generation QorIQ SoCs.

The WRITE and READ bandwidth improvement is achieved through target queue enhancements; specifically, changes to the scheduling algorithm, improvements in the bank hashing scheme, support for more transaction re-ordering, and additional proprietary techniques.

4.8 Universal serial bus (USB) 2.0

The two USB 2.0 controllers with integrated PHY provide point-to-point connectivity that complies with the USB specification, Rev. 2.0. Each of the USB controllers with integrated PHY can be configured to operate as a stand-alone host, and one of the controllers (USB #2) can be configured as a stand-alone device, or with both host and device functions operating simultaneously.

Key features of the USB 2.0 controller include the following:

- Complies with USB specification, Rev. 2.0
- Supports high-speed (480 Mbps), full-speed (12 Mbps), and low-speed (1.5 Mbps) operations
- Both controllers support operation as a stand-alone USB host controller
  - Supports USB root hub with one downstream-facing port
  - Enhanced host controller interface (EHCI)-compatible
- Both controllers supports operation as a stand-alone USB device
  - Support one upstream-facing port
  - Support six programmable USB endpoints

The host and device functions are both configured to support all four USB transfer types:

- Bulk
- Control
- Interrupt
- Isochronous

4.9 High-speed peripheral interface complex (HSSI)

The SoC offers a variety of high-speed serial interfaces, sharing a set of 16 SerDes lanes. Each interface is backed by a high speed serial interface controller. The SoC has the following types and quantities of controllers:

- Four PCI Express controllers, one Gen 3.0 PCI Express controller with SRIOV, one Gen 3.0 PCI Express controller without SRIOV and two PCI Express Gen 3.0 controllers
- Two Serial RapidIO 2.0
- Two SATA 2.0
- Up to eight Ethernet controllers with various protocols
- Aurora

The features of each high-speed serial controller are described in the subsequent sections. Debug functionality is described in Debug support."

4.9.1 PCI Express

This chip instantiates four PCI Express controllers, each with the following key features:

- One PCI Express controller supports end-point SR-IOV.
  - Two physical functions
  - 64 virtual functions per physical function
  - Eight MSI-X per either physical function or virtual function
- Two PCI Express controllers support 2.0 (maximum lane width off x8).
- Two PCI Express controllers support 3.0 (maximum lane width of x4).
- Power-on reset configuration options allow root complex or endpoint functionality.
- x8, x4, x2, and x1 link widths support
- Both 32- and 64-bit addressing and 256-byte maximum payload size
- Inbound INTx transactions
- Message signaled interrupt (MSI) transactions

4.9.2 Serial RapidIO

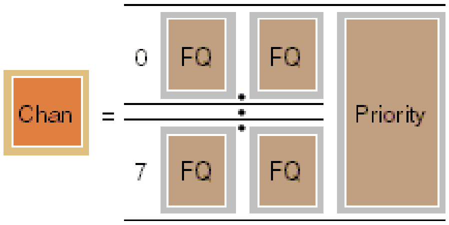
The Serial RapidIO interface is based on the *RapidIO Interconnect Specification, Revision 2.1* . RapidIO is a high-performance, point-to-point, low-pin-count, packet-switched system-level interconnect that can be used in a variety of applications as an open standard. The rich feature set includes high data bandwidth, low-latency capability, and support for high-performance I/O devices as well as message-passing and software-managed programming models. Receive and transmit ports operate independently, and with 2 x 4 Serial RapidIO controllers, the aggregate theoretical bandwidth is 32 Gbps.

The chip offers two Serial RapidIO controllers Receive and transmit ports operate independently and with 2 x 4 Serial RapidIO controllers; the aggregate theoretical bandwidth is 32 Gbps. The Serial RapidIO controllers can be used in conjunction with "Rapid IO Message Manager (RMAN), as described in RapidIO Message Manager (RMan 1.0)."

Key features of the Serial RapidIO interface unit include the following:

- Support for *RapidIO Interconnect Specification, Revision 2.1* (All transaction flows and priorities.)

Table 3. DPAA terms and definitions (continued)

Term	Definition	Graphic representation
Channel	Set of eight WQs with hardware provided prioritized access	
Dedicated channel	Channel statically assigned to a particular end point, from which that end point can dequeue frames. End point may be a CPU, FMan, PME, or SEC.	-
Pool channel	A channel statically assigned to a group of end points, from which any of the end points may dequeue frames.	

4.10.2 Major DPAA components

The SoC's Datapath Acceleration Architecture, shown in the figure below, includes the following major components:

- Frame Manager (FMan)
- Queue Manager (QMan)
- Buffer Manager (BMan)
- RapidIO Message Manager (RMan 1.0)
- Security Engine (SEC 5.2)
- Pattern Matching Engine (PME 2.1)
- Decompression and Compression Engine (DCE 1.0)

The QMan and BMan are infrastructure components, which are used by both software and hardware for queuing and memory allocation/deallocation. The Frame Managers and RMan are interfaces between the external world and the DPAA. These components receive datagrams via Ethernet or Serial RapidIO and queue them to other DPAA entities, as well as dequeue datagrams from other DPAA entities for transmission. The SEC, PME, and DCE are content accelerators that dequeue processing requests (typically from software) and enqueue results to the configured next consumer. Each component is described in more detail in the following sections.

- Common interface between software and all hardware
    - Controls the prioritized queuing of data between multiple processor cores, network interfaces, and hardware accelerators.
    - Supports both dedicated and pool channels, allowing both push and pull models of multicore load spreading.
- Atomic access to common queues without software locking overhead
- Mechanisms to guarantee order preservation with atomicity and order restoration following parallel processing on multiple CPUs
- Egress queuing for Ethernet interfaces
    - Hierarchical (2-level) scheduling and dual-rate shaping
    - Dual-rate shaping to meet service-level agreements (SLAs) parameters (1 Kbps...10 Gbps range, 1 Kbps granularity across the entire range)
    - Configurable combinations of strict priority and fair scheduling (weighted queuing) between the queues
    - Algorithms for shaping and fair scheduling are based on bytes
- Queuing to cores and accelerators
    - Two level queuing hierarchy with one or more Channels per Endpoint, eight work queues per Channel, and numerous frame queues per work queue
    - Priority and work conserving fair scheduling between the work queues and the frame queues
- Loss-less flow control for ingress network interfaces
- Congestion avoidance (RED/WRED) and congestion management with tail discard

4.10.2.3 Buffer Manager

The Buffer Manager (BMan) manages pools of buffers on behalf of software for both hardware (accelerators and network interfaces) and software use.

The Buffer Manager offers the following features:

- Common interface for software and hardware
- Guarantees atomic access to shared buffer pools
- Supports 64 buffer pools
    - Software, hardware buffer consumers can request different size buffers and buffers in different memory partitions
- Supports depletion thresholds with congestion notifications
- On-chip per pool buffer stockpile to minimize access to memory for buffer pool management
- LIFO (last in first out) buffer allocation policy
    - Optimizes cache usage and allocation
    - A released buffer is immediately used for receiving new data

4.10.2.4 Pattern Matching Engine (PME 2.1)

The PME 2.1 is Freescale's second generation of extended NFA style pattern matching engine. Unchanged from the first generation QorIQ products, it supports ~10 Gbps data scanning.

Key benefits of a NFA pattern matching engine:

- No pattern "explosion" to support "wildcarding" or case-insensitivity
    - Comparative compilations have shown 300,000 DFA pattern equivalents can be achieved with ~8000 extended NFA patterns
- Pattern density much higher than DFA engines.
    - Patterns can be stored in on-chip tables and main DDR memory
    - Most work performed solely with on-chip tables (external memory access required only to confirm a match)
    - No need for specialty memories; for example, QDR SRAM, RLDRAM, and so on.
- Fast compilation of pattern database, with fast incremental additions
    - Pattern database can be updated without halting processing
    - Only affected pattern records are downloaded
    - DFA style engines can require minutes to hours to recompile and compress database

Freescale's basic NFA capabilities for byte pattern scanning are as follows:

- The PME's regex compiler accepts search patterns using syntax similar to that in software-based regex engines, such as Perl-Compatible Regular Expression (PCRE).
  - Supports Perl meta-characters including wildcards, repeats, ranges, anchors, and so on.
  - Byte patterns are simple matches, such as gabcd123h, existing in both the data being scanned and in the pattern specification database.
- Up to 32 KB patterns of length 1-128 bytes

Freescale's extensions to NFA style pattern matching are principally related to event pattern scanning. Event patterns are sequences of byte patterns linked by 'stateful rules.' Freescale uses event pattern scanning and stateful rule processing synonymously. Stateful rules are hardware instructions by which users define reactions to pattern match events, such as state changes, assignments, bitwise operations, addition, subtraction, and comparisons.

Some key characteristics and benefits of the Stateful Rule extensions include:

- Ability to match patterns across data "work units" or packet boundaries
  - Can be used to correlate patterns, qualify matches (for example, contextual match), or to track protocol state change
- Easily support "greedy" wildcards
  - For example, ABC.*DEF == two patterns tied together by a stateful rule
- Delays the need for software post-processing. Software is alerted after all byte patterns are detected in the proper sequence, rather than any time a byte pattern is detected.
- Implements a significant subset of the regex pattern definition syntax as well as many constructs which cannot be expressed in standard PCRE
- PME 2.1 supports up to 32K stateful rules, linking multiple byte patterns

The PME 2.1 dequeues data from its QMan hardware portal and, based on FQ configuration, scans the data against one of 256 pattern sets, 16 subsets per pattern set.

When the PME finds a byte pattern match, or a final pattern in a stateful rule, it generates a report.

4.10.2.5 RapidIO Message Manager (RMan 1.0)

The RapidIO message manager (RMan) produces and consumes Type 8 Port-write, Type 9 Data Streaming, Type 10 Doorbells and Type 11 Messaging traffic and is capable of producing Type 5 NWRITE and Type 6 SWRITE transactions. For inbound traffic, the RMan supports up to 17 open reassembly contexts as a arbitrary mix of Type 9, and Type 11 traffic.

As ingress packets arrive at the RMan, they are compared against up to 64 classification rules to determine the target queue. These rules support Type 8, 9, 10 and 11 transaction types. They may be wild-carded and are configured as masks over selected header fields. This table lists the fields that are maskable as part of each classification rule.

Table 4. Maskable fields in each classification rule

Classification rule	Field
Transaction types	RapidIO port
	Source device ID
	Destination device ID
	Flow level
Type 9 messaging-specific	Class-of-service (CoS)
	StreamID
Type 11 messaging-specific	Mailbox
	Extended mailbox
	Letter

- Public-key hardware accelerators (PKHA)
    - RSA and Diffie-Hellman (to 4096b)
    - Elliptic curve cryptography (1023b)
- Data-encryption standard accelerators (DESA)
    - DES, 3DES (2-key, 3-key)
    - ECB, CBC, OFB, and CFB modes
- Advanced-encryption standard accelerators (AESA)
    - Key lengths of 128-bit, 192-bit, and 256-bit
    - Confidentiality modes
        - ECB, CBC, OFB, CFB, CTR and XTS
    - Authenticated encryption modes
        - CCM and GCM
- ARC four hardware accelerators (AFHA)
    - Compatible with RC4 algorithm
- Message digest hardware accelerators (MDHA)
    - SHA-1, SHA-256, 384, 512-bit digests
    - MD5 128-bit digest
    - HMAC with all algorithms
- Kasumi/F8 hardware accelerators (KFHA)
    - F8, F9 as required for 3GPP
    - A5/3 for GSM and EDGE, GEA-3 for GPRS
- Snow 3G hardware accelerators (STHA)
    - Implements Snow 3.0, F8 and F9 modes
- ZUC Hardware Accelerators (ZHA)
    - Implements 128-EEA3 & 128-EIA3
- CRC Unit
    - Standard and user-defined polynomials
- Random-number generator (RNG)
    - Incorporates TRNG entropy generator for seeding and deterministic engine (SHA-256)
    - Supports random IV generation
- DTLS
- IEEE Std 802.11 WiFi

Protocol	Cipher suite	Performance (aggregate encap and decap)
IPsec	AES-CBC/AES-XCBC-MAC	4.4 Gbps
LTE PDCP U-plane	128-EEA2 (AES)	8.8 Gbps
LTE PDCP C-plane	128-EEA3 and 128-EIA3 (ZUC)	3.5 Gbps

The SEC dequeues data from its QMan hardware portal and, based on FQ configuration, also dequeues associated instructions and operands in the Shared Descriptor. The SEC processes the data then enqueues it to the configured output FQ. The SEC uses the Status/CMD word in the output Frame Descriptor to inform the next consumer of any errors encountered during processing (for example, received packet outside the anti-replay window.)

The SEC 5.2 is also part of the QorIQ Platform's Trust Architecture, which gives the SoC the ability to perform secure boot, runtime code integrity protection, and session key protection. The Trust Architecture is described in Resource partitioning and QorIQ Trust Architecture.

4.10.2.7 Decompression and Compression Engine (DCE 1.0)

The Decompression and Compression Engine (DCE 1.0) is an accelerator compatible with Datapath Architecture providing lossless data decompression and compression for the QorIQ family of SoCs. The DCE supports the raw DEFLATE algorithm (RFC1951), GZIP format (RFC1952) and ZLIB format (RFC1950). The DCE also supports Base 64 encoding and decoding (RFC4648).

The DEFLATE algorithm is a basic building block for data compression in most modern communication systems. It is used by HTTP to compress web pages, by SSL to compress records, by gzip to compress files and email attachments, and by many other applications.

Deflate involves searching for repeated patterns previously seen in a Frame, computing the length and the distance of the pattern with respect to the current location in the Frame, and encoding the resulting information into a bitstream.

The decompression algorithm involves decoding the bitstream and replaying past data. The Decompression and Compression Engine is architected to minimize the system memory bandwidth required to do decompression and compression of Frames while providing multi-gigabits per second of performance.

Detailed features include the following:

- Deflate; as specified as in RFC1951
- GZIP; as specified in RFC1952
- Zlib; as specified in RFC1950
  - Interoperable with the zlib 1.2.5 compression library
- Compression
  - ZLIB, GZIP and DEFLATE header insertion
  - ZLIB and GZIP CRC computation and insertion
  - Zlib sync flush and partial flush for chunked compression (for example, for HTTP1.1)
  - Four modes of compression
    - No compression (just add DEFLATE header)
    - Encode only using static/dynamic Huffman codes
    - Compress and encode using static Huffman codes
    - Compress and encode using dynamic Huffman codes
  - Uses a 4KB sliding history window
  - Supports Base 64 encoding (RFC4648) after compression
  - Provides at least 2.5:1 compression ratio on the Calgary Corpus
- Decompression supports:
  - ZLIB, GZIP and DEFLATE header removal
  - ZLIB and GZIP CRC validation
  - 32KB history
  - Zlib flush for chunked decompression (for HTTP1.1 for example)
    - All standard modes of decompression
    - No compression
    - Static Huffman codes
    - Dynamic Huffman codes
  - Provides option to return original compressed Frame along with the uncompressed Frame or release the buffers to BMan
  - Does not support use of ZLIB preset dictionaries (FDICT flag = 1 is treated as an error).
  - Base 64 decoding (RFC4648) prior to decompression

The DCE 1.0 is designed to support up to 8.8 Gbps for either compression or decompression, or 17.5 Gbps aggregate at ~4 KB data sizes.

4.11 OCeaN DMA

The OCeaN fabric is used to:

- Transfer general data between two memory locations

- Eight high-speed/high-bandwidth channels
- Basic DMA operation modes (direct, simple chaining)
- Extended DMA operation modes (advanced chaining and stride capability)
- Programmable bandwidth control between channels
- Three priority levels supported for source and destination transactions
- Can be activated using DREQ pin
- Optimized to work with the high speed interfaces
- Address translation and mapping unit (ATMU) which allows to define packet attributes as address/device/flow level/transaction type. ATMU Bypass that allows the descriptor to specify the attributes.

4.12 Serial memory controllers

In addition to the parallel NAND and NOR flash supported by means of the IFC, the chip supports serial flash using eSPI and SD/eSDHC/eMMC card and device interfaces. The SD/eSDHC/eMMC controller includes a DMA engine, allowing it to move data from serial flash to external or internal memory following straightforward initiation by software.

Detailed features of the eSPI controller include the following:

- Supports SPI full-duplex or half-duplex single master mode with four independent chip-selects
- Supports RapidS™ full clock cycle operation, and Winbond dual output read
- Independent, programmable baud-rate generator
- Programmable clock phase and polarity
- Supports four different configurations per chip-select

Detailed features of the SD/eSDHC/eMMC controller include the following:

- Conforms to the *SD Host Controller Standard Specification version 3.0*
- Compatible with the *MMC System Specification version 4.5*
- Compatible with the *SD Memory Card Physical Layer Specification version 3.01*
- Compatible with the *SD - SDIO Card Specification version 2.0*
- Designed to work with eMMC devices as well as SD Memory, SDIO, and SD Combo cards and their variants
- Supports SD UHS-1 speed modes

4.12.1 PreBoot Loader and nonvolatile memory interfaces

The PreBoot Loader (PBL) operates on behalf of a large number of interfaces.

4.12.1.1 PreBoot Loader (PBL)

The PBL's functions include the following:

- Simplifies boot operations, replacing pin strapping resistors with configuration data loaded from nonvolatile memory
- Uses the configuration data to initialize other system logic and to copy data from low speed memory interfaces (I^2C, IFC, eSPI, SD/SDXC/eMMC) into fully initialized DDR or other access targets in the chip
- Releases CPU 0 from reset, allowing the boot processes to begin from fast system memory

4.12.1.2 Integrated Flash Controller

The SoC incorporates an Integrated Flash Controller similar to the one used in some previous generation QorIQ SoCs. The IFC supports both NAND and NOR flash, as well as a general purpose memory mapped interface for connecting low speed ASICs and FPGAs.

4.12.1.2.1 NAND Flash features

- x8/x16 NAND Flash interface
- Optional ECC generation/checking
- Flexible timing control to allow interfacing with proprietary NAND devices
- SLC and MLC Flash devices support with configurable page sizes of up to 8 KB
- Support advance NAND commands like cache, copy-back, and multiplane programming
- Boot chip-select (CS0) available after system reset, with boot block size of 8 KB, for execute-in-place boot loading from NAND Flash
- Up to terabyte Flash devices supported

4.12.1.2.2 NOR Flash features

- Data bus width of 8/16
- Compatible with asynchronous NOR Flash
- Directly memory mapped
- Supports address data multiplexed (ADM) NOR device
- Flexible timing control allows interfacing with proprietary NOR devices
- Boot chip-select (CS0) available at system reset

4.12.1.2.3 General-purpose chip-select machine (GPCM)

The IFC's GPCM supports the following features:

- Normal GPCM
  - Support for x8/16-bit device
  - Compatible with general purpose addressable device, for example, SRAM and ROM
  - External clock is supported with programmable division ratio (2, 3, 4, and so on, up to 16)
- Generic ASIC Interface
  - Support for x8/16-bit device
  - Address and Data are shared on I/O bus
  - Following address and data sequences are supported on I/O bus:
    - 16-bit I/O: AADD
    - 8-bit I/O: AAAADDDD

4.13 Resource partitioning and QorIQ Trust Architecture

Consolidation of discrete CPUs into a single, multicore chip introduces many opportunities for unintended resource contentions to arise, particularly when multiple, independent software entities reside on a single chip. A system may exhibit erratic behavior if multiple software partitions cannot effectively partition resources. Device consolidation, combined with a trend toward embedded systems becoming more open (or more likely to run third-party or open-source software on at least one of the cores), creates opportunities for malicious code to enter a system.

This chip offers a new level of hardware partitioning support, allowing system developers to ensure software running on any CPU only accesses the resources (memory, peripherals, and so on) that it is explicitly authorized to access. This section provides an overview of the features implemented in the chip that help ensure that only trusted software executes on the CPUs, and that the trusted software remains in control of the system with intended isolation.

4.13.1 Core MMU, UX/SX bits, and embedded hypervisor

The chip's first line of defense against unintended interactions amongst the multiple CPUs/OSes is each core vCPU's MMU. A vCPU's MMU is configured to determine which addresses in the global address map the CPU is able to read or write. If a particular resource (memory region, peripheral device, and so on) is dedicated to a single vCPU, that vCPU's MMU is configured to allow access to those addresses (on 4 KB granularity); other vCPU MMUs are not configured for access to those addresses, which makes them private. When two vCPUs need to share resources, their MMUs are both configured so that they have access to the shared address range.

This level of hardware support for partitioning is common today; however, it is not sufficient for many core systems running diverse software. When the functions of multiple discrete CPUs are consolidated onto a single multicore chip, achieving strong partitioning should not require the developer to map functions onto vCPUs that are the exclusive owners of specific platform resources. The alternative, a fully open system with no private resources, is also unacceptable. For this reason, the core's MMU also includes three levels of access permissions: user, supervisor (OS), and hypervisor. An embedded hypervisor (for example, KVM, XEN, QorIQ ecosystem partner hypervisor) runs unobtrusively beneath the various OSes running on the vCPUs, consuming CPU cycles only when an access attempt is made to an embedded hypervisor-managed shared resource.

The embedded hypervisor determines whether the access should be allowed and, if so, proxies the access on behalf of the original requestor. If malicious or poorly tested software on any vCPU attempts to overwrite important device configuration registers (including vCPU's MMU), the embedded hypervisor blocks the write. High and low-speed peripheral interfaces (PCI Express, UART), when not dedicated to a single vCPU/partition, are other examples of embedded hypervisor managed resources. The degree of security policy enforcement by the embedded hypervisor is implementation-dependent.

In addition to defining regions of memory as being controlled by the user, supervisor, or hypervisor, the core MMU can also configure memory regions as being non-executable. Preventing CPUs from executing instructions from regions of memory used as data buffers is a powerful defense against buffer overflows and other runtime attacks. In previous generations of Power Architecture, this feature was controlled by the NX (no execute) attribute. In new Power Architecture cores such as the e6500 core, there are separate bits controlling execution for user (UX) and supervisor (SX).

4.13.2 Peripheral access management unit (PAMU)

MMU-based access control works for software running on CPUs; however, these are not the only bus masters in the SoC. Internal components with bus mastering capability (FMan, RMan, PCI Express controller, PME, SEC, and so on) also need to be prevented from reading and writing to certain memory regions. These components do not spontaneously generate access attempts; however, if programmed to do so by buggy or malicious software, any of them could read or write sensitive data registers and crash the system. For this reason, the SoC also includes a distributed function referred to as the peripheral access management unit (PAMU).

PAMUs provide address translation and access control for all non-CPU initiators in the system. PAMU access control is based on the logical I/O device number (LIODN) advertised by a bus master for a given transaction. LIODNs can be static (for example, PCI Express controller #1 always uses LIODN 123) or they can be dynamic, based on the ID of the CPU that programmed the initiator (for example, the SEC uses LIODN 456 because it was given a descriptor by vCPU #2). In the dynamic example, the SoC architecture provides positive identification of the vCPU programming the SEC, preventing LIODN spoofing.

4.13.3 IO partitioning

The simplest IO configuration in chips running multiple independent software partitions is to dedicate specific IO controllers (PCI Express, SATA, Serial RapidIO controllers) to specific vCPUs. The core MMUs and PAMUs can enforce these access permissions to insure that only the software partition owning the IO is able to use it. The obvious problem with this approach is that there are likely to be more software partitions wanting IO access than there are IO controllers to dedicate to each.

Safe IO sharing can be accomplished through the use of a hypervisor; however, there is a performance penalty associated with virtual IO, as the hypervisor must consume CPU cycles to schedule the IO requests and get the results back to the right software partition.

The DPAA (described in Data Path Acceleration Architecture (DPAA)") was designed to allow multiple partitions to efficiently share accelerators and IOs, with its major capabilities centered around sharing Ethernet ports. These capabilities were enhanced in the chip with the addition of FMan storage profiles. The chip's FMans perform classification prior to buffer pool selection, allowing Ethernet frames arriving on a single port to be written to the dedicated memory of a single software partition. This capability is fully described in Receiver functionality: parsing, classification, and distribution."

The addition of the RMan extends the chip's IO virtualization by allowing many types of traffic arriving on Serial RapidIO to enter the DPAA and take advantage of its inherent virtualization and partitioning capabilities.

The PCI Express protocol lacks the PDU semantics found in Serial RapidIO, making it difficult to interwork between PCI Express controllers and the DPAA; however, PCI Express has made progress in other areas of partition. The Single Root IO Virtualization specification, which the chip supports as an endpoint, allows external hosts to view the chip as multiple four physical functions (PFs), where each PF supports up to 64 virtual functions (VFs). Having multiple VFs on a PCI Express port effectively channelizes it, so that each transaction through the port is identified as belonging to a specific PF/VF combination (with associated and potentially dedicated memory regions). Message signalled interrupts (MSIs) allow the external Host to generate interrupts associated with a specific VF.

4.13.4 Secure boot and sensitive data protection

The core MMUs and PAMU allow the SoC to enforce a consistent set of memory access permissions on a per-partition basis. When combined with an embedded hypervisor for safe sharing of resources, the SoC becomes highly resilient to poorly tested or malicious code. For system developers building high reliability/high security platforms, rigorous testing of code of known origin is the norm.

For this reason, the SoC offers a secure boot option, in which the system developer digitally signs the code to be executed by the CPUs, and the SoC insures that only an unaltered version of that code runs on the platform. The SoC offers both boot time and run time code authenticity checking, with configurable consequences when the authenticity check fails. The SoC also supports protected internal and external storage of developer-provisioned sensitive instructions and data. For example, a system developer may provision each system with a number of RSA private keys to be used in mutual authentication and key exchange. These values would initially be stored as encrypted blobs in external non-volatile memory; but, following secure boot, these values can be decrypted into on-chip protected memory (portion of platform cache dedicated as SRAM). Session keys, which may number in the thousands to tens of thousands, are not good candidates for on-chip storage, so the SoC offers session key encryption. Session keys are stored in main memory, and are decrypted (transparently to software and without impacting SEC throughput) as they are brought into the for decryption of session traffic.

4.14 Advanced power management

Power dissipation is always a major design consideration in embedded applications; system designers need to balance the desire for maximum compute and IO density against single-chip and board-level thermal limits.

Advances in chip and board level cooling have allowed many OEMs to exceed the traditional 30 W limit for a single chip, and Freescale's flagship T4240 multicore chip, has consequently retargeted its maximum power dissipation. A top-speed bin T4240 dissipates approximately 2x the power dissipation of the P4080; however, the T4240 increases computing performance by ~4x, yielding a 2x improvement in DMIPs per watt.

Junction temperature is a critical factor in comparing embedded processor specifications. Freescale specs max power at 105C junction, standard for commercial, embedded operating conditions. Not all multicore chips adhere to a 105C junction for specifying worst case power. In the interest of normalizing power comparisons, the chip's typical and worst case power (all CPUs at 1.8 GHz) are shown at alternate junction temperatures.

To achieve the previously-stated 2x increase in performance per watt, the chip implements a number of software transparent and performance transparent power management features. Non-transparent power management features are also available, allowing for significant reductions in power consumption when the chip is under lighter loads; however, non-transparent power savings are not assumed in chip power specifications.

4.14.1 Transparent power management

This chip's commitment to low power begins with the decision to fabricate the chip in 28 nm bulk CMOS. This process technology offers low leakage, reducing both static and dynamic power. While 28 nm offers inherent power savings, transistor leakage varies from lot to lot and device to device. Leakier parts are capable of faster transistor switching, but they also consume more power. By running devices from the leakier end of the process spectrum at less than nominal voltage and devices from the slower end of the process spectrum at higher nominal voltage, T2080-based systems can achieve the required operating frequency within the specified max power. During manufacturing, Freescale will determine the voltage required to achieve the target frequency bin and program this Voltage ID into each device, so that initialization software can program the system's voltage regulator to the appropriate value.

Dynamic power is further reduced through fine-grained clock control. Many components and subcomponents in the chip automatically sleep (turn off their clocks) when they are not actively processing data. Such blocks can return to full operating frequency on the clock cycle after work is dispatched to them. A portion of these dynamic power savings are built into the chip max power specification on the basis of impossibility of all processing elements and interfaces in the chip switching concurrently. The percent switching factors are considered quite conservative, and measured typical power consumption on QorIQ chips is well below the maximum in the data sheet.

As noted in Frame Manager and network interfaces, the chip supports Energy-Efficient Ethernet. During periods of extended inactivity on the transmit side, the chip transparently sends a low power idle (LPI) signal to the external PHY, effectively telling it to sleep.

Additional power savings can be achieved by users statically disabling unused components. Developers can turn off the clocks to individual logic blocks (including CPUs) within the chip that the system is not using. Based on a finite number of SerDes, it is expected that any given application will have some inactive Ethernet MACs, PCI Express, or serial RapidIO controllers. Re-enabling clocks to a logic block generally requires an chip reset, which makes this type of power management infrequent (effectively static) and transparent to runtime software.

4.14.2 Non-transparent power management

Many load-based power savings are use-case specific static configurations (thereby software transparent), and were described in the previous section. This section focuses on SoC power management mechanisms, which software can dynamically leverage to reduce power when the system is lightly loaded. The most important of these mechanisms involves the cores.

A full description of core low-power states with proper names is provided in the SoC reference manual. At a high level, the most important of these states can be viewed as "PH10" and "PH20," described as follows. Note that these are relative terms, which do not perfectly correlate to previous uses of these terms in Power Architecture and other ISAs:

- In PH10 state CPU stops instruction fetches but still performs L1 snoops. The CPU retains all state, and instruction fetching can be restarted instantly.
- In PH20 state CPU stops instruction fetches and L1 snooping, and turns off all clocks. Supply voltage is reduced, using a technique Freescale calls State Retention Power Gating (SRPG). In the "napping" state, a CPU uses ~75% less power than a fully operational CPU, but can still return to full operation quickly (~100 platform clocks).

The core offers two ways to enter these (and other) low power states: registers and instructions.

As the name implies, register-based power management means that software writes to registers to select the CPU and its low power state. Any CPU with write access to power management registers can put itself, or another CPU, into a low power state; however, a CPU put into a low power state by way of register write cannot wake itself up.

Instruction-based power management means that software executes special WAIT instruction to enter a low power state. CPUs exit the low power state in response to external triggers, interrupts, doorbells, stashes into L1-D cache, or clear reservation on snoop. Each vCPU can independently execute WAIT instructions; however, the physical CPU enters PH20 state after the second vCPU executes its wait. The instruction-based "enters PH20 state" state is particularly well-suited for use in conjunction with Freescale's patented Cascade Power Management, which is described in the next section.

While significant power savings can be achieved through individual CPU low power states, the SoC also supports a register-based cluster level low power state. After software puts all CPUs in a cluster in a PH10 state, it can additionally flush the L2 cache and have the entire cluster enter PH20 state . Because the L2 arrays have relatively low static power dissipation, this state provides incremental additional savings over having four napping CPUs with the L2 on.

4.14.3 Cascade power management

Cascade power management refers to the concept of allowing SoC load, as defined by the depth of queues managed by the Queue Manager, to determine how many vCPUs need to be awake to handle the load. Recall from Queue Manager that the QMan supports both dedicated and pool channels. Pool channels are channels of frame queues consumed by parallel workers (vCPUs), where any worker can process any packet dequeued from the channel.

Cascade Power Management exploits the QMan's awareness of vCPU membership in a pool channel and overall pool channel queue depth. The QMan uses this information to tell vCPUs in a pool channel (starting with the highest numbered vCPU) that they can execute instructions to enter PH10 mode. When pool channel queue depth exceeds configurable thresholds, the QMan wakes up the lowest numbered vCPU.

The SoC's dynamic power management capabilities, whether using the Cascade scheme or a master control CPU and load to power matching software, enable up to a 75% reduction in power consumption versus data sheet max power.

4.15 Debug support

The reduced number of external buses enabled by the move to multicore chips greatly simplifies board level lay-out and eliminates many concerns over signal integrity. While the board designer may embrace multicore CPUs, software engineers have real concerns over the potential to lose debug visibility.

Processing on a multicore chip with shared caches and peripherals also leads to greater concurrency and an increased potential for unintended interactions between device components. To ensure that software developers have the same or better visibility into the device as they would with multiple discrete communications processors, Freescale developed an Advanced Multicore Debug Architecture.

The debugging and performance monitoring capability enabled by the device hardware coexists within a debug ecosystem that offers a rich variety of tools at different levels of the hardware/software stack. Software development and debug tools from Freescale (CodeWarrior), as well as third-party vendors, provide a rich set of options for configuring, controlling, and analyzing debug and performance related events.

Appendix A T2081

A.1 Introduction

The T2081 QorIQ advanced, multicore processor combines four, dual-threaded e6500 Power Architecture® processor cores with high-performance datapath acceleration logic and network and peripheral bus interfaces required for networking, telecom/datacom, wireless infrastructure, and mil/aerospace applications.

This figure shows the major functional units within the chip.

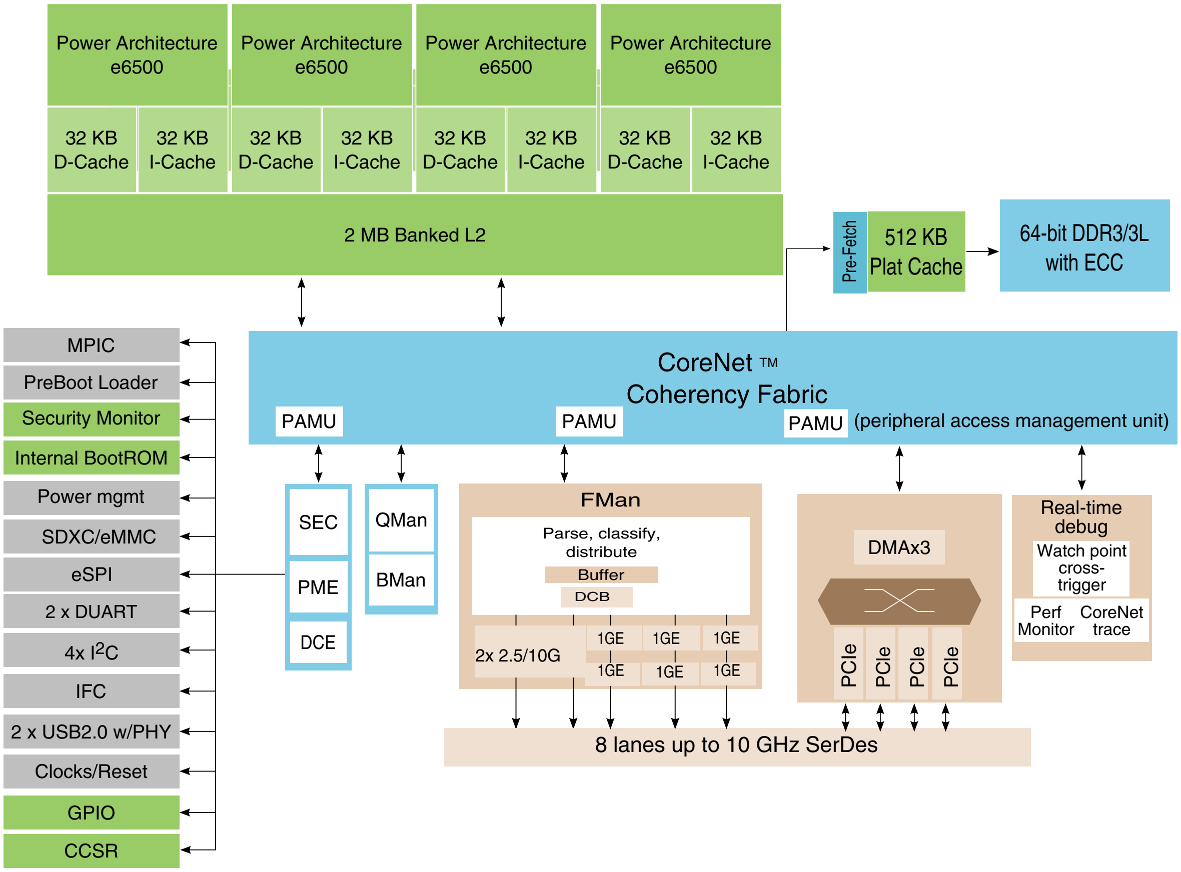

Power Architecture e6500
32 KB D-Cache
32 KB I-Cache
2 MB Banked L2
Pre-Fetch
512 KB Plat Cache
64-bit DDR3/3L with ECC
CoreNet ™ Coherency Fabric
PAMU
PAMU
PAMU (peripheral access management unit)
MPIC
PreBoot Loader
Security Monitor
Internal BootROM
Power mgmt
SDXC/eMMC
eSPI
2 x DUART
4x I2C
IFC
2 x USB2.0 w/PHY
Clocks/Reset
GPIO
CCSR
SEC
PME
DCE
QMan
BMan
FMan
Parse, classify, distribute
Buffer
DCB
2x 2.5/10G
1GE
1GE
1GE
1GE
1GE
1GE
DMAx3
PCIe
PCIe
PCIe
PCIe
Real-time debug
Watch point cross-trigger
Perf Monitor
CoreNet trace
8 lanes up to 10 GHz SerDes

Figure A-1. T2081 block diagram

A.2 Overview of Differences

Table A-1. Comparison between T2080 and T2081

Feature	T2080	T2081
Peripherals		
10G Ethernet Controllers	Up to four with XFI, 10GBase-KR, 10GBase-KX, XAUI, HiGig and HiGig2	Up to two XFI or 10GBase-KR, 10GBase-KX
1G Ethernet Controllers	Up to eight	Up to six
SerDes and Pinout		
Total number of SerDes lanes	16	8
High Speed Serial IO		
SRIO Controller and RapidIO Message Manager	2 + RMan	not supported
SATA Controller	2	not supported
Aurora	supported	not supported
Package	25 x 25mm, 896 pins, 0.8mm pitch	23 x 23mm, 780 pins, 0.8mm pitch, pin compatible with T1042

How to Reach Us:

Home Page:
freescale.com

Web Support:
freescale.com/support

Information in this document is provided solely to enable system and software implementers to use Freescale products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. Freescale reserves the right to make changes without further notice to any products herein.

Freescale makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Freescale assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. "Typical" parameters that may be provided in Freescale data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including "typicals," must be validated for each customer application by customer's technical experts. Freescale does not convey any license under its patent rights nor the rights of others. Freescale sells products pursuant to standard terms and conditions of sale, which can be found at the following address: freescale.com/SalesTermsandConditions.

Freescale, the Freescale logo, AltiVec, CodeWarrior, and QorIQ are trademarks of Freescale Semiconductor, Inc., Reg. U.S. Pat. & Tm. Off. CoreNet is a trademark of Freescale Semiconductor, Inc. All other product or service names are the property of their respective owners. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

© 2014 Freescale Semiconductor, Inc.

Document Number T2080PB
Revision 0, 04/2014